UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/994,516	11/26/2001	Trung T. Doan	500966.01	8536

7590 05/26/2005

Kimton N. Eng, Esq.
DORSEY & WHITNEY LLP
1420 Fifth Avenue, Suite 3400
Seattle, WA 98101

EXAMINER	
CLEARY, THOMAS J	
ART UNIT	PAPER NUMBER
2111	

DATE MAILED: 05/26/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/994,516	DOAN ET AL.	
	Examiner	Art Unit	
	Thomas J. Cleary	2111	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☐ Responsive to communication(s) filed on *03 March 2005*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-50* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-50* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date *20041122,20050328*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☒ Other: *IDS Paper No(s) 20050425*.

DETAILED ACTION

Claim Rejections - 35 USC § 103

1. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

2. Claims 1, 2, 4, 5, 6, 7, 8, 9, 11, 12, 13, 14, 15 16, 17, 19, 20, 21, 22, 23, 25, 26, 27, 38, 39, 40, 41, 42, 44, 45, 46, 47, 48, and 50 are rejected under 35 U.S.C. 103(a) as being unpatentable over PCI System Architecture, Third Edition, by Tom Shanley ("Shanley"), US Patent Number 6,098,158 to Lay et al. ("Lay"), and Intel Application Note AP-758 'Flash Memory PCI Add-In Card for Embedded Systems' ("AP-758").

3. In reference to Claim 1, Shanley teaches a central processing unit (CPU) (See Figure 2-3 'CPU'); a first bus coupled to the CPU (See Figure 2-3 'CPU Local Bus'); a memory coupled to the first bus to store data accessible by the CPU via the first bus (See Figure 2-3 'Main Memory'); a second bus coupled to the first bus to provide communication with the CPU and the memory via the first bus (See Figure 2-3 'PCI Bus'); and a PC card coupled to the second bus (See Figure 2-3). Shanley does not teach the PC card having a non-volatile memory for storing machine state information

and further having a controller coupled to the non-volatile memory for coordinating with the CPU access to the non-volatile memory and the memory to store and download the machine state information for capturing and restoring, respectively, a corresponding machine state of a computer system. Lay teaches storing machine state information in a non-volatile memory (See Figure 4 and Column 2 Line 23 – Column 3 Line 5) and a controller coupled to the non-volatile memory for coordinating with the CPU access to the non-volatile memory and the memory to store and download the machine state information for capturing and restoring, respectively, a corresponding machine state of a computer system (See Column 2 Lines 26-31, Column 2 Lines 45-54, Column 3 Lines 2-5, and Column 6 Lines 36-43). AP-758 teaches a placing a non-volatile memory on a PC card (See Page 1 Section 1.0).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the computer system of Shanley with the fast boot by saving the system state to non-volatile memory of Lay in order to allow the system to boot up faster (See Column 1 Lines 39-50 and Column 2 Lines 23-40 of Lay). It would have been further obvious to place the non-volatile memory on the PC card of AP-758 in order to allow the system to function as an embedded system (See Page 1 Section 1.0 Paragraph 1 of AP-758), because the PCI bus presents an easy to implement interface that can use current chipset technologies from the host processor bus, has good performance, industry standard protocols, and provisions for system boot control (See Page 1 Section 1.0 Paragraph 5 of AP-758), and because it can be easily

integrated into existing systems that support the PCI bus (See Page 1 Section 1.0 Paragraph 9 of AP-758).

4. In reference to Claim 2, Shanley, AP-758, and Lay teach the limitations as applied to Claim 1 above. Shanley further teaches that the first bus comprises a local CPU bus (See Figure 2-3 'CPU Local Bus') and the second bus comprises a PCI bus (See Figure 2-3 'PCI Bus').

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the computer system of Shanley with the fast boot by saving the system state to non-volatile memory of Lay in order to allow the system to boot up faster (See Column 1 Lines 39-50 and Column 2 Lines 23-40 of Lay). It would have been further obvious to place the non-volatile memory on the PC card of AP-758 in order to allow the system to function as an embedded system (See Page 1 Section 1.0 Paragraph 1 of AP-758), because the PCI bus presents an easy to implement interface that can use current chipset technologies from the host processor bus, has good performance, industry standard protocols, and provisions for system boot control (See Page 1 Section 1.0 Paragraph 5 of AP-758), and because it can be easily integrated into existing systems that support the PCI bus (See Page 1 Section 1.0 Paragraph 9 of AP-758).

5. In reference to Claim 4, Shanley, AP-758, and Lay teach the limitations as applied to Claim 1 above. AP-758 further teaches that the non-volatile memory of the PC card comprises a flash memory device (See Page 1 Section 1.0 Paragraphs 2-7).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the computer system of Shanley with the fast boot by saving the system state to non-volatile memory of Lay in order to allow the system to boot up faster (See Column 1 Lines 39-50 and Column 2 Lines 23-40 of Lay). It would have been further obvious to place the non-volatile memory on the PC card of AP-758 in order to allow the system to function as an embedded system (See Page 1 Section 1.0 Paragraph 1 of AP-758), because the PCI bus presents an easy to implement interface that can use current chipset technologies from the host processor bus, has good performance, industry standard protocols, and provisions for system boot control (See Page 1 Section 1.0 Paragraph 5 of AP-758), and because it can be easily integrated into existing systems that support the PCI bus (See Page 1 Section 1.0 Paragraph 9 of AP-758).

6. In reference to Claim 5, Shanley, AP-758, and Lay teach the limitations as applied to Claim 1 above. AP-758 further teaches that the PC card further includes a bus interface coupled to the second bus, and further coupled to the non-volatile memory and the controller to transfer data between the non-volatile memory and the second bus in accordance with a data format and transfer protocol of the second bus. (See Page 2 Figure 2 and Page 3 Paragraph 2).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the computer system of Shanley with the fast boot by saving the system state to non-volatile memory of Lay in order to allow the system to boot up faster (See Column 1 Lines 39-50 and Column 2 Lines 23-40 of Lay). It would have been further obvious to place the non-volatile memory on the PC card of AP-758 in order to allow the system to function as an embedded system (See Page 1 Section 1.0 Paragraph 1 of AP-758), because the PCI bus presents an easy to implement interface that can use current chipset technologies from the host processor bus, has good performance, industry standard protocols, and provisions for system boot control (See Page 1 Section 1.0 Paragraph 5 of AP-758), and because it can be easily integrated into existing systems that support the PCI bus (See Page 1 Section 1.0 Paragraph 9 of AP-758).

7. In reference to Claim 6, Shanley, AP-758, and Lay teach the limitations as applied to Claim 1 above. Lay further teaches a transfer component directing the controller to coordinate access between the non-volatile memory and the memory to transfer machine state information (See Figures 4 and 6, Column 2 Lines 26-31, Column 2 Lines 45-54, Column 3 Lines 2-5, and Column 5 Lines 39-63).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the computer system of Shanley with the fast boot by saving the system state to non-volatile memory of Lay in order to allow the system to boot up faster (See Column 1 Lines 39-50 and Column 2 Lines 23-40 of Lay). It would

have been further obvious to place the non-volatile memory on the PC card of AP-758 in order to allow the system to function as an embedded system (See Page 1 Section 1.0 Paragraph 1 of AP-758), because the PCI bus presents an easy to implement interface that can use current chipset technologies from the host processor bus, has good performance, industry standard protocols, and provisions for system boot control (See Page 1 Section 1.0 Paragraph 5 of AP-758), and because it can be easily integrated into existing systems that support the PCI bus (See Page 1 Section 1.0 Paragraph 9 of AP-758).

8. In reference to Claim 7, Shanley, AP-758, and Lay teach the limitations as applied to Claim 1 above. Shanley, AP-758, and Lay do not teach compression and decompression components for compressing the machine state information to be stored and decompressing the stored compressed machine state information to be downloaded, respectively. Lay, however, teaches that it is well known to compress an image of the machine state information to be saved to disk, and to later decompress the image when booting the system (See Column 1 Lines 51-60).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the computer system of Shanley with the fast boot by saving the system state to non-volatile memory and the compression and decompression of the boot image of Lay in order to allow the system to boot up faster (See Column 1 Lines 39-50 and Column 2 Lines 23-40 of Lay). It would have been further obvious to place the non-volatile memory on the PC card of AP-758 in order to

allow the system to function as an embedded system (See Page 1 Section 1.0 Paragraph 1 of AP-758), because the PCI bus presents an easy to implement interface that can use current chipset technologies from the host processor bus, has good performance, industry standard protocols, and provisions for system boot control (See Page 1 Section 1.0 Paragraph 5 of AP-758), and because it can be easily integrated into existing systems that support the PCI bus (See Page 1 Section 1.0 Paragraph 9 of AP-758). It would have been further obvious to compress the image of the machine state because it is well known that compressed data takes up less space in memory than uncompressed data, thus allowing more efficient use of the memory space.

9. In reference to Claim 8, Shanley, AP-758, and Lay teach the limitations as applied to Claim 1 above. Lay further teaches that the machine state information comprises data from the memory and CPU for returning the computer system to the same condition of operability as when the machine state information was stored in the non-volatile memory (See Figure 3 and Column 2 Lines 26-37).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the computer system of Shanley with the fast boot by saving the system state to non-volatile memory of Lay in order to allow the system to boot up faster (See Column 1 Lines 39-50 and Column 2 Lines 23-40 of Lay). It would have been further obvious to place the non-volatile memory on the PC card of AP-758 in order to allow the system to function as an embedded system (See Page 1 Section 1.0 Paragraph 1 of AP-758), because the PCI bus presents an easy to implement

interface that can use current chipset technologies from the host processor bus, has good performance, industry standard protocols, and provisions for system boot control (See Page 1 Section 1.0 Paragraph 5 of AP-758), and because it can be easily integrated into existing systems that support the PCI bus (See Page 1 Section 1.0 Paragraph 9 of AP-758).

10. In reference to Claim 9, Shanley teaches a central processing unit (CPU) (See Figure 2-3 'CPU'); a memory coupled to the CPU to store data accessible by the CPU (See Figure 2-3 'Main Memory'); a bus coupled to the CPU and the memory to provide communication therewith (See Figure 2-3 'PCI Bus'); and a PC card coupled to the bus (See Figure 2-3). Shanley does not teach the PC card having a non-volatile memory for storing machine state information and further having a controller coupled to the non-volatile memory for coordinating with the CPU access to the non-volatile memory and the memory to store and download the machine state information for capturing and restoring, respectively, a corresponding machine state of a computer system. AP-758 teaches a PC card having a non-volatile memory (See Page 1 Section 1.0). Lay teaches storing machine state information in a non-volatile memory (See Figure 4 and Column 2 Line 23 – Column 3 Line 5) and a controller coupled to the non-volatile memory for coordinating with the CPU access to the non-volatile memory and the memory to store and download the machine state information for capturing and restoring, respectively, a corresponding machine state of a computer system (See Column 2 Lines 26-31, Column 2 Lines 45-54, and Column 3 Lines 2-5).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the computer system of Shanley with the fast boot by saving the system state to non-volatile memory of Lay in order to allow the system to boot up faster (See Column 1 Lines 39-50 and Column 2 Lines 23-40 of Lay). It would have been further obvious to place the non-volatile memory on the PC card of AP-758 in order to allow the system to function as an embedded system (See Page 1 Section 1.0 Paragraph 1 of AP-758), because the PCI bus presents an easy to implement interface that can use current chipset technologies from the host processor bus, has good performance, industry standard protocols, and provisions for system boot control (See Page 1 Section 1.0 Paragraph 5 of AP-758), and because it can be easily integrated into existing systems that support the PCI bus (See Page 1 Section 1.0 Paragraph 9 of AP-758).

11. In reference to Claim 11, Shanley, AP-758, and Lay teach the limitations as applied to Claim 9 above. AP-758 further teaches that the non-volatile memory of the PC card comprises a flash memory device (See Page 1 Section 1.0 Paragraphs 2-7).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the computer system of Shanley with the fast boot by saving the system state to non-volatile memory of Lay in order to allow the system to boot up faster (See Column 1 Lines 39-50 and Column 2 Lines 23-40 of Lay). It would have been further obvious to place the non-volatile memory on the PC card of AP-758 in order to allow the system to function as an embedded system (See Page 1 Section

1.0 Paragraph 1 of AP-758), because the PCI bus presents an easy to implement interface that can use current chipset technologies from the host processor bus, has good performance, industry standard protocols, and provisions for system boot control (See Page 1 Section 1.0 Paragraph 5 of AP-758), and because it can be easily integrated into existing systems that support the PCI bus (See Page 1 Section 1.0 Paragraph 9 of AP-758).

12. In reference to Claim 12, Shanley, AP-758, and Lay teach the limitations as applied to Claim 9 above. AP-758 further teaches that the PC card further includes a bus interface coupled to the bus, and further coupled to the non-volatile memory and the controller to transfer data between the non-volatile memory and the bus in accordance with a data format and transfer protocol of the bus. (See Page 2 Figure 2 and Page 3 Paragraph 2).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the computer system of Shanley with the fast boot by saving the system state to non-volatile memory of Lay in order to allow the system to boot up faster (See Column 1 Lines 39-50 and Column 2 Lines 23-40 of Lay). It would have been further obvious to place the non-volatile memory on the PC card of AP-758 in order to allow the system to function as an embedded system (See Page 1 Section 1.0 Paragraph 1 of AP-758), because the PCI bus presents an easy to implement interface that can use current chipset technologies from the host processor bus, has good performance, industry standard protocols, and provisions for system boot control

(See Page 1 Section 1.0 Paragraph 5 of AP-758), and because it can be easily integrated into existing systems that support the PCI bus (See Page 1 Section 1.0 Paragraph 9 of AP-758).

13. In reference to Claim 13, Shanley, AP-758, and Lay teach the limitations as applied to Claim 9 above. Lay further teaches a transfer component directing the controller to coordinate access between the non-volatile memory and the memory to transfer machine state information (See Figures 4 and 6, Column 2 Lines 26-31, Column 2 Lines 45-54, Column 3 Lines 2-5, and Column 5 Lines 39-63).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the computer system of Shanley with the fast boot by saving the system state to non-volatile memory of Lay in order to allow the system to boot up faster (See Column 1 Lines 39-50 and Column 2 Lines 23-40 of Lay). It would have been further obvious to place the non-volatile memory on the PC card of AP-758 in order to allow the system to function as an embedded system (See Page 1 Section 1.0 Paragraph 1 of AP-758), because the PCI bus presents an easy to implement interface that can use current chipset technologies from the host processor bus, has good performance, industry standard protocols, and provisions for system boot control (See Page 1 Section 1.0 Paragraph 5 of AP-758), and because it can be easily integrated into existing systems that support the PCI bus (See Page 1 Section 1.0 Paragraph 9 of AP-758).

14. In reference to Claim 14, Shanley, AP-758, and Lay teach the limitations as applied to Claim 9 above. Shanley, AP-758, and Lay do not teach compression and decompression components for compressing the machine state information to be stored and decompressing the stored compressed machine state information to be downloaded, respectively. Lay, however, teaches that it is well known to compress an image of the machine state information to be saved to disk, and to later decompress the image when booting the system (See Column 1 Lines 51-60).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the computer system of Shanley with the fast boot by saving the system state to non-volatile memory and the compression and decompression of the boot image of Lay in order to allow the system to boot up faster (See Column 1 Lines 39-50 and Column 2 Lines 23-40 of Lay). It would have been further obvious to place the non-volatile memory on the PC card of AP-758 in order to allow the system to function as an embedded system (See Page 1 Section 1.0 Paragraph 1 of AP-758), because the PCI bus presents an easy to implement interface that can use current chipset technologies from the host processor bus, has good performance, industry standard protocols, and provisions for system boot control (See Page 1 Section 1.0 Paragraph 5 of AP-758), and because it can be easily integrated into existing systems that support the PCI bus (See Page 1 Section 1.0 Paragraph 9 of AP-758). It would have been further obvious to compress the image of the machine state because it is well known that compressed data takes up less space in memory than uncompressed data, thus allowing more efficient use of the memory space.

15. In reference to Claim 15, Shanley, AP-758, and Lay teach the limitations as applied to Claim 9 above. Lay further teaches that the machine state information comprises data from the memory and CPU for returning the computer system to the same condition of operability as when the machine state information was stored in the non-volatile memory (See Figure 3 and Column 2 Lines 26-37).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the computer system of Shanley with the fast boot by saving the system state to non-volatile memory of Lay in order to allow the system to boot up faster (See Column 1 Lines 39-50 and Column 2 Lines 23-40 of Lay). It would have been further obvious to place the non-volatile memory on the PC card of AP-758 in order to allow the system to function as an embedded system (See Page 1 Section 1.0 Paragraph 1 of AP-758), because the PCI bus presents an easy to implement interface that can use current chipset technologies from the host processor bus, has good performance, industry standard protocols, and provisions for system boot control (See Page 1 Section 1.0 Paragraph 5 of AP-758), and because it can be easily integrated into existing systems that support the PCI bus (See Page 1 Section 1.0 Paragraph 9 of AP-758).

16. In reference to Claim 16, Shanley teaches an apparatus having a central processing unit (CPU) (See Figure 2-3 'CPU') coupled to a memory (See Figure 2-3 'Main Memory') via a first bus (See Figure 2-3 'CPU Local Bus'), and further having a

second bus coupled to the first bus to provide communication with the CPU and the memory (See Figure 2-3 'PCI Bus'); and a PC card coupled to the second bus (See Figure 2-3). Shanley does not teach that the apparatus is for capturing and restoring a machine state of a computer system, the PC card having a non-volatile memory for storing machine state information corresponding to the machine state, and further having a controller coupled to the non-volatile memory to control the storing of data therein and the retrieval of data therefrom; and a transfer component for directing the controller to coordinate with the CPU access to the non-volatile memory and the memory to store and download the machine state information for capturing and restoring, respectively, a corresponding machine state of a computer system. AP-758 teaches a PC card having a non-volatile memory (See Page 1 Section 1.0). Lay teaches storing machine state information in a non-volatile memory (See Figure 4 and Column 2 Line 23 – Column 3 Line 5); a controller coupled to the non-volatile memory to control the storing of data therein and the retrieval of data therefrom (See Column 2 Lines 26-31, Column 2 Lines 45-54, and Column 3 Lines 2-5); and a transfer component directing the controller to coordinate with the CPU access to the non-volatile memory and the memory to store and download the machine state information for capturing and restoring, respectively, a corresponding machine state of a computer system (See Figures 4 and 6, Column 2 Lines 26-31, Column 2 Lines 45-54, Column 3 Lines 2-5, and Column 5 Lines 39-63).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the computer system of Shanley with the fast boot by

saving the system state to non-volatile memory of Lay in order to allow the system to boot up faster (See Column 1 Lines 39-50 and Column 2 Lines 23-40 of Lay). It would have been further obvious to place the non-volatile memory on the PC card of AP-758 in order to allow the system to function as an embedded system (See Page 1 Section 1.0 Paragraph 1 of AP-758), because the PCI bus presents an easy to implement interface that can use current chipset technologies from the host processor bus, has good performance, industry standard protocols, and provisions for system boot control (See Page 1 Section 1.0 Paragraph 5 of AP-758), and because it can be easily integrated into existing systems that support the PCI bus (See Page 1 Section 1.0 Paragraph 9 of AP-758).

17. In reference to Claim 17, Shanley, AP-758, and Lay teach the limitations as applied to Claim 16 above. AP-758 further teaches that the PC card further includes a bus interface coupled to the second bus, and further coupled to the non-volatile memory and the controller to transfer data between the non-volatile memory and the second bus in accordance with a data format and transfer protocol of the second bus. (See Page 2 Figure 2 and Page 3 Paragraph 2).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the computer system of Shanley with the fast boot by saving the system state to non-volatile memory of Lay in order to allow the system to boot up faster (See Column 1 Lines 39-50 and Column 2 Lines 23-40 of Lay). It would have been further obvious to place the non-volatile memory on the PC card of AP-758

in order to allow the system to function as an embedded system (See Page 1 Section 1.0 Paragraph 1 of AP-758), because the PCI bus presents an easy to implement interface that can use current chipset technologies from the host processor bus, has good performance, industry standard protocols, and provisions for system boot control (See Page 1 Section 1.0 Paragraph 5 of AP-758), and because it can be easily integrated into existing systems that support the PCI bus (See Page 1 Section 1.0 Paragraph 9 of AP-758).

18. In reference to Claim 19, Shanley, AP-758, and Lay teach the limitations as applied to Claim 16 above. AP-758 further teaches that the non-volatile memory comprises a flash memory (See Page 1 Section 1.0 Paragraphs 2-7).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the computer system of Shanley with the fast boot by saving the system state to non-volatile memory of Lay in order to allow the system to boot up faster (See Column 1 Lines 39-50 and Column 2 Lines 23-40 of Lay). It would have been further obvious to place the non-volatile memory on the PC card of AP-758 in order to allow the system to function as an embedded system (See Page 1 Section 1.0 Paragraph 1 of AP-758), because the PCI bus presents an easy to implement interface that can use current chipset technologies from the host processor bus, has good performance, industry standard protocols, and provisions for system boot control (See Page 1 Section 1.0 Paragraph 5 of AP-758), and because it can be easily

integrated into existing systems that support the PCI bus (See Page 1 Section 1.0 Paragraph 9 of AP-758).

19. In reference to Claim 20, Shanley, AP-758, and Lay teach the limitations as applied to Claim 16 above. Lay further teaches the transfer component comprises: a storing component for directing the controller to store machine state information from the CPU and memory to the non-volatile memory (See Figure 3, Column 2 Lines 26-31, and Column 2 Lines 45-54); and a download component for directing the controller to transfer data from the nonvolatile memory to the CPU and the memory (See Figure 5, Column 3 Lines 2-5, and Column 5 Lines 32-63).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the computer system of Shanley with the fast boot by saving the system state to non-volatile memory of Lay in order to allow the system to boot up faster (See Column 1 Lines 39-50 and Column 2 Lines 23-40 of Lay). It would have been further obvious to place the non-volatile memory on the PC card of AP-758 in order to allow the system to function as an embedded system (See Page 1 Section 1.0 Paragraph 1 of AP-758), because the PCI bus presents an easy to implement interface that can use current chipset technologies from the host processor bus, has good performance, industry standard protocols, and provisions for system boot control (See Page 1 Section 1.0 Paragraph 5 of AP-758), and because it can be easily integrated into existing systems that support the PCI bus (See Page 1 Section 1.0 Paragraph 9 of AP-758).

20. In reference to Claim 21, Shanley, AP-758, and Lay teach the limitations as applied to Claim 16 above. Shanley, AP-758, and Lay do not teach compression and decompression components for compressing the machine state information to be stored and decompressing the stored compressed machine state information to be downloaded, respectively. Lay, however, teaches that it is well known to compress an image of the machine state information to be saved to disk, and to later decompress the image when booting the system (See Column 1 Lines 51-60).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the computer system of Shanley with the fast boot by saving the system state to non-volatile memory and the compression and decompression of the boot image of Lay in order to allow the system to boot up faster (See Column 1 Lines 39-50 and Column 2 Lines 23-40 of Lay). It would have been further obvious to place the non-volatile memory on the PC card of AP-758 in order to allow the system to function as an embedded system (See Page 1 Section 1.0 Paragraph 1 of AP-758), because the PCI bus presents an easy to implement interface that can use current chipset technologies from the host processor bus, has good performance, industry standard protocols, and provisions for system boot control (See Page 1 Section 1.0 Paragraph 5 of AP-758), and because it can be easily integrated into existing systems that support the PCI bus (See Page 1 Section 1.0 Paragraph 9 of AP-758). It would have been further obvious to compress the image of the machine

state because it is well known that compressed data takes up less space in memory than uncompressed data, thus allowing more efficient use of the memory space.

21. In reference to Claim 22, Shanley teaches an apparatus having a central processing unit (CPU) (See Figure 2-3 'CPU') coupled to a memory (See Figure 2-3 'Main Memory'), and further having a bus coupled to the CPU and memory to provide communication with the CPU and the memory (See Figure 2-3 'PCI Bus'); and a PC card coupled to the second bus (See Figure 2-3). Shanley does not teach that the apparatus is for capturing and restoring a machine state of a computer system, the PC card having a non-volatile memory for storing machine state information corresponding to the machine state, and further having a controller coupled to the non-volatile memory to control the storing of data therein and the retrieval of data therefrom; and a transfer component for directing the controller to coordinate with the CPU access to the non-volatile memory and the memory to store and download the machine state information for capturing and restoring, respectively, a corresponding machine state of a computer system. AP-758 teaches a PC card having a non-volatile memory (See Page 1 Section 1.0). Lay teaches storing machine state information in a non-volatile memory (See Figure 4 and Column 2 Line 23 – Column 3 Line 5); a controller coupled to the non-volatile memory to control the storing of data therein and the retrieval of data therefrom (See Column 2 Lines 26-31, Column 2 Lines 45-54, and Column 3 Lines 2-5); and a transfer component directing the controller to coordinate with the CPU access to the non-volatile memory and the memory to store and download the machine state

information for capturing and restoring, respectively, a corresponding machine state of a computer system (See Figures 4 and 6, Column 2 Lines 26-31, Column 2 Lines 45-54, Column 3 Lines 2-5, and Column 5 Lines 39-63).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the computer system of Shanley with the fast boot by saving the system state to non-volatile memory of Lay in order to allow the system to boot up faster (See Column 1 Lines 39-50 and Column 2 Lines 23-40 of Lay). It would have been further obvious to place the non-volatile memory on the PC card of AP-758 in order to allow the system to function as an embedded system (See Page 1 Section 1.0 Paragraph 1 of AP-758), because the PCI bus presents an easy to implement interface that can use current chipset technologies from the host processor bus, has good performance, industry standard protocols, and provisions for system boot control (See Page 1 Section 1.0 Paragraph 5 of AP-758), and because it can be easily integrated into existing systems that support the PCI bus (See Page 1 Section 1.0 Paragraph 9 of AP-758).

22. In reference to Claim 23, Shanley, AP-758, and Lay teach the limitations as applied to Claim 22 above. AP-758 further teaches that the PC card further includes a bus interface coupled to the bus, and further coupled to the non-volatile memory and the controller to transfer data between the non-volatile memory and the bus in accordance with a data format and transfer protocol of the second bus. (See Page 2 Figure 2 and Page 3 Paragraph 2).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the computer system of Shanley with the fast boot by saving the system state to non-volatile memory of Lay in order to allow the system to boot up faster (See Column 1 Lines 39-50 and Column 2 Lines 23-40 of Lay). It would have been further obvious to place the non-volatile memory on the PC card of AP-758 in order to allow the system to function as an embedded system (See Page 1 Section 1.0 Paragraph 1 of AP-758), because the PCI bus presents an easy to implement interface that can use current chipset technologies from the host processor bus, has good performance, industry standard protocols, and provisions for system boot control (See Page 1 Section 1.0 Paragraph 5 of AP-758), and because it can be easily integrated into existing systems that support the PCI bus (See Page 1 Section 1.0 Paragraph 9 of AP-758).

23. In reference to Claim 25, Shanley, AP-758, and Lay teach the limitations as applied to Claim 23 above. AP-758 further teaches that the non-volatile memory comprises a flash memory (See Page 1 Section 1.0 Paragraphs 2-7).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the computer system of Shanley with the fast boot by saving the system state to non-volatile memory of Lay in order to allow the system to boot up faster (See Column 1 Lines 39-50 and Column 2 Lines 23-40 of Lay). It would have been further obvious to place the non-volatile memory on the PC card of AP-758 in order to allow the system to function as an embedded system (See Page 1 Section

1.0 Paragraph 1 of AP-758), because the PCI bus presents an easy to implement interface that can use current chipset technologies from the host processor bus, has good performance, industry standard protocols, and provisions for system boot control (See Page 1 Section 1.0 Paragraph 5 of AP-758), and because it can be easily integrated into existing systems that support the PCI bus (See Page 1 Section 1.0 Paragraph 9 of AP-758).

24. In reference to Claim 26, Shanley, AP-758, and Lay teach the limitations as applied to Claim 23 above. Lay further teaches the transfer component comprises: a storing component for directing the controller to store machine state information from the CPU and memory to the non-volatile memory (See Figure 3, Column 2 Lines 26-31, and Column 2 Lines 45-54); and a download component for directing the controller to transfer data from the nonvolatile memory to the CPU and the memory (See Figure 5, Column 3 Lines 2-5, and Column 5 Lines 32-63).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the computer system of Shanley with the fast boot by saving the system state to non-volatile memory of Lay in order to allow the system to boot up faster (See Column 1 Lines 39-50 and Column 2 Lines 23-40 of Lay). It would have been further obvious to place the non-volatile memory on the PC card of AP-758 in order to allow the system to function as an embedded system (See Page 1 Section 1.0 Paragraph 1 of AP-758), because the PCI bus presents an easy to implement interface that can use current chipset technologies from the host processor bus, has

good performance, industry standard protocols, and provisions for system boot control (See Page 1 Section 1.0 Paragraph 5 of AP-758), and because it can be easily integrated into existing systems that support the PCI bus (See Page 1 Section 1.0 Paragraph 9 of AP-758).

25. In reference to Claim 27, Shanley, AP-758, and Lay teach the limitations as applied to Claim 23 above. Shanley, AP-758, and Lay do not teach compression and decompression components for compressing the machine state information to be stored and decompressing the stored compressed machine state information to be downloaded, respectively. Lay, however, teaches that it is well known to compress an image of the machine state information to be saved to disk, and to later decompress the image when booting the system (See Column 1 Lines 51-60).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the computer system of Shanley with the fast boot by saving the system state to non-volatile memory and the compression and decompression of the boot image of Lay in order to allow the system to boot up faster (See Column 1 Lines 39-50 and Column 2 Lines 23-40 of Lay). It would have been further obvious to place the non-volatile memory on the PC card of AP-758 in order to allow the system to function as an embedded system (See Page 1 Section 1.0 Paragraph 1 of AP-758), because the PCI bus presents an easy to implement interface that can use current chipset technologies from the host processor bus, has good performance, industry standard protocols, and provisions for system boot control (See

Page 1 Section 1.0 Paragraph 5 of AP-758), and because it can be easily integrated into existing systems that support the PCI bus (See Page 1 Section 1.0 Paragraph 9 of AP-758). It would have been further obvious to compress the image of the machine state because it is well known that compressed data takes up less space in memory than uncompressed data, thus allowing more efficient use of the memory space.

26. In reference to Claim 38, Shanley teaches a computer system having a central processing unit (CPU) (See Figure 2-3 'CPU') coupled to a memory (See Figure 2-3 'Main Memory'), and further having a bus coupled to the CPU and memory to provide communication therewith (See Figure 2-3 'PCI Bus'), and a PC card (See Figure 2-3). Shanley does not teach a method for storing a machine state of the computer system, comprising: capturing the machine state of the computer system via a controller coupled to a non-volatile memory to control the storing of data therein and the retrieval of data therefrom; transferring machine state information corresponding to the captured machine state from the computer system to a PC card operably coupled with the non-volatile memory; and storing the machine state information in the non-volatile memory in order to restore the stored machine state when the machine state information is provided to a computer system. AP-758 teaches a PC card operably coupled with a non-volatile memory (See Page 1 Section 1.0). Lay teaches capturing the machine state of the computer system via a controller coupled to a non-volatile memory to control the storing of data therein and the retrieval of data therefrom (See Column 2 Lines 26-37); transferring machine state information corresponding to the captured

machine state from the computer system to a non-volatile memory (See Figure 4 and Column 2 Line 23 – Column 3 Line 5); and storing the machine state information in the non-volatile memory in order to restore the stored machine state when the machine state information is provided to a computer system (See Figure 4 and Column 2 Line 23 – Column 3 Line 5).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the computer system of Shanley with the fast boot by saving the system state to non-volatile memory of Lay in order to allow the system to boot up faster (See Column 1 Lines 39-50 and Column 2 Lines 23-40 of Lay). It would have been further obvious to place the non-volatile memory on the PC card of AP-758 in order to allow the system to function as an embedded system (See Page 1 Section 1.0 Paragraph 1 of AP-758), because the PCI bus presents an easy to implement interface that can use current chipset technologies from the host processor bus, has good performance, industry standard protocols, and provisions for system boot control (See Page 1 Section 1.0 Paragraph 5 of AP-758), and because it can be easily integrated into existing systems that support the PCI bus (See Page 1 Section 1.0 Paragraph 9 of AP-758).

27. In reference to Claim 39, Shanley, AP-758, and Lay teach the limitations as applied to Claim 38 above. Lay further teaches that capturing, transferring and storing the machine state information is in response to executing a power down procedure (See Column 4 Lines 20-25).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the computer system of Shanley with the fast boot by saving the system state to non-volatile memory of Lay in order to allow the system to boot up faster (See Column 1 Lines 39-50 and Column 2 Lines 23-40 of Lay). It would have been further obvious to place the non-volatile memory on the PC card of AP-758 in order to allow the system to function as an embedded system (See Page 1 Section 1.0 Paragraph 1 of AP-758), because the PCI bus presents an easy to implement interface that can use current chipset technologies from the host processor bus, has good performance, industry standard protocols, and provisions for system boot control (See Page 1 Section 1.0 Paragraph 5 of AP-758), and because it can be easily integrated into existing systems that support the PCI bus (See Page 1 Section 1.0 Paragraph 9 of AP-758).

28. In reference to Claim 40, Shanley, AP-758, and Lay teach the limitations as applied to Claim 38 above. Lay further teaches that capturing, transferring and storing the machine state information is in response to a user request (See Column 4 Lines 20-25, Column 4 Lines 41-55, and Column 5 Lines 36-38).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the computer system of Shanley with the fast boot by saving the system state to non-volatile memory of Lay in order to allow the system to boot up faster (See Column 1 Lines 39-50 and Column 2 Lines 23-40 of Lay). It would have been further obvious to place the non-volatile memory on the PC card of AP-758

in order to allow the system to function as an embedded system (See Page 1 Section 1.0 Paragraph 1 of AP-758), because the PCI bus presents an easy to implement interface that can use current chipset technologies from the host processor bus, has good performance, industry standard protocols, and provisions for system boot control (See Page 1 Section 1.0 Paragraph 5 of AP-758), and because it can be easily integrated into existing systems that support the PCI bus (See Page 1 Section 1.0 Paragraph 9 of AP-758).

29. In reference to Claim 41, Shanley, AP-758, and Lay teach the limitations as applied to Claim 38 above. Lay further teaches that the machine state information comprises data from the memory and CPU for returning the computer system to the same condition of operability as when the machine state information was stored in the non-volatile memory (See Figure 3 and Column 2 Lines 26-37).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the computer system of Shanley with the fast boot by saving the system state to non-volatile memory of Lay in order to allow the system to boot up faster (See Column 1 Lines 39-50 and Column 2 Lines 23-40 of Lay). It would have been further obvious to place the non-volatile memory on the PC card of AP-758 in order to allow the system to function as an embedded system (See Page 1 Section 1.0 Paragraph 1 of AP-758), because the PCI bus presents an easy to implement interface that can use current chipset technologies from the host processor bus, has good performance, industry standard protocols, and provisions for system boot control

(See Page 1 Section 1.0 Paragraph 5 of AP-758), and because it can be easily integrated into existing systems that support the PCI bus (See Page 1 Section 1.0 Paragraph 9 of AP-758).

30. In reference to Claim 42, Shanley, AP-758, and Lay teach the limitations as applied to Claim 38 above. Lay further teaches that capturing the machine state of the computer system comprises: capturing data present in the memory (See Column 2 Lines 33-37); and capturing data present in registers of the CPU (See Column 5 Lines 1-6).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the computer system of Shanley with the fast boot by saving the system state to non-volatile memory of Lay in order to allow the system to boot up faster (See Column 1 Lines 39-50 and Column 2 Lines 23-40 of Lay). It would have been further obvious to place the non-volatile memory on the PC card of AP-758 in order to allow the system to function as an embedded system (See Page 1 Section 1.0 Paragraph 1 of AP-758), because the PCI bus presents an easy to implement interface that can use current chipset technologies from the host processor bus, has good performance, industry standard protocols, and provisions for system boot control (See Page 1 Section 1.0 Paragraph 5 of AP-758), and because it can be easily integrated into existing systems that support the PCI bus (See Page 1 Section 1.0 Paragraph 9 of AP-758).

31. In reference to Claim 44, Shanley, AP-758, and Lay teach the limitations as applied to Claim 38 above. Shanley, AP-758, and Lay do not teach compressing the machine state information to be stored in the non-volatile memory. Lay, however, teaches that it is well known to compress an image of the machine state information to be saved to disk, and to later decompress the image when booting the system (See Column 1 Lines 51-60).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the computer system of Shanley with the fast boot by saving the system state to non-volatile memory and the compression and decompression of the boot image of Lay in order to allow the system to boot up faster (See Column 1 Lines 39-50 and Column 2 Lines 23-40 of Lay). It would have been further obvious to place the non-volatile memory on the PC card of AP-758 in order to allow the system to function as an embedded system (See Page 1 Section 1.0 Paragraph 1 of AP-758), because the PCI bus presents an easy to implement interface that can use current chipset technologies from the host processor bus, has good performance, industry standard protocols, and provisions for system boot control (See Page 1 Section 1.0 Paragraph 5 of AP-758), and because it can be easily integrated into existing systems that support the PCI bus (See Page 1 Section 1.0 Paragraph 9 of AP-758). It would have been further obvious to compress the image of the machine state because it is well known that compressed data takes up less space in memory than uncompressed data, thus allowing more efficient use of the memory space.

32. In reference to Claim 45, Shanley teaches a computer system having a central processing unit (CPU) (See Figure 2-3 'CPU') coupled to a memory (See Figure 2-3 'Main Memory'), and further having a bus coupled to the CPU and memory to provide communication therewith (See Figure 2-3 'PCI Bus'), and a PC card (See Figure 2-3). Shanley does not teach a method for restoring a machine state of the computer system, comprising: identifying machine state information corresponding to the machine state to which the computer system is to be restored stored in a non-volatile memory included in a PC card; transferring the machine state information from the non-volatile memory to the computer system via a controller coupled to the non-volatile memory to control the storing of data therein and the retrieval of data therefrom; and writing data of the machine state information to the memory and CPU in order to restore the computer system to the identified machine state. AP-758 teaches a PC card having a non-volatile memory (See Page 1 Section 1.0). Lay teaches identifying machine state information corresponding to the machine state to which the computer system is to be restored stored in a non-volatile memory (See Column 5 Lines 41-44) ; transferring the machine state information from the non-volatile memory to the computer system via a controller coupled to the non-volatile memory to control the storing of data therein and the retrieval of data therefrom (See Figure 6 and Column 5 Lines 44-46); and writing data of the machine state information to the memory and CPU in order to restore the computer system to the identified machine state (See Figures 5 and 6 and Column 5 Lines 53-63).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the computer system of Shanley with the fast boot by

saving the system state to non-volatile memory of Lay in order to allow the system to boot up faster (See Column 1 Lines 39-50 and Column 2 Lines 23-40 of Lay). It would have been further obvious to place the non-volatile memory on the PC card of AP-758 in order to allow the system to function as an embedded system (See Page 1 Section 1.0 Paragraph 1 of AP-758), because the PCI bus presents an easy to implement interface that can use current chipset technologies from the host processor bus, has good performance, industry standard protocols, and provisions for system boot control (See Page 1 Section 1.0 Paragraph 5 of AP-758), and because it can be easily integrated into existing systems that support the PCI bus (See Page 1 Section 1.0 Paragraph 9 of AP-758).

33. In reference to Claim 46, Shanley, AP-758, and Lay teach the limitations as applied to Claim 45 above. Lay further teaches that identifying, transferring and writing the machine state information is in response to executing a power up procedure (See Column 4 Lines 20-25 and Column 5 Lines 32-34).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the computer system of Shanley with the fast boot by saving the system state to non-volatile memory of Lay in order to allow the system to boot up faster (See Column 1 Lines 39-50 and Column 2 Lines 23-40 of Lay). It would have been further obvious to place the non-volatile memory on the PC card of AP-758 in order to allow the system to function as an embedded system (See Page 1 Section 1.0 Paragraph 1 of AP-758), because the PCI bus presents an easy to implement

interface that can use current chipset technologies from the host processor bus, has good performance, industry standard protocols, and provisions for system boot control (See Page 1 Section 1.0 Paragraph 5 of AP-758), and because it can be easily integrated into existing systems that support the PCI bus (See Page 1 Section 1.0 Paragraph 9 of AP-758).

34. In reference to Claim 47, Shanley, AP-758, and Lay teach the limitations as applied to Claim 45 above. Lay further teaches that identifying, transferring and writing the machine state information is in response to a user request (See Column 4 Lines 20-25, Column 4 Lines 41-55, and Column 5 Lines 36-38).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the computer system of Shanley with the fast boot by saving the system state to non-volatile memory of Lay in order to allow the system to boot up faster (See Column 1 Lines 39-50 and Column 2 Lines 23-40 of Lay). It would have been further obvious to place the non-volatile memory on the PC card of AP-758 in order to allow the system to function as an embedded system (See Page 1 Section 1.0 Paragraph 1 of AP-758), because the PCI bus presents an easy to implement interface that can use current chipset technologies from the host processor bus, has good performance, industry standard protocols, and provisions for system boot control (See Page 1 Section 1.0 Paragraph 5 of AP-758), and because it can be easily integrated into existing systems that support the PCI bus (See Page 1 Section 1.0 Paragraph 9 of AP-758).

35. In reference to Claim 48, Shanley, AP-758, and Lay teach the limitations as applied to Claim 45 above. Lay further teaches that the machine state information comprises data from the memory and CPU for returning the computer system to the same condition of operability as when the machine state information was stored in the non-volatile memory (See Figure 3 and Column 2 Lines 26-37).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the computer system of Shanley with the fast boot by saving the system state to non-volatile memory of Lay in order to allow the system to boot up faster (See Column 1 Lines 39-50 and Column 2 Lines 23-40 of Lay). It would have been further obvious to place the non-volatile memory on the PC card of AP-758 in order to allow the system to function as an embedded system (See Page 1 Section 1.0 Paragraph 1 of AP-758), because the PCI bus presents an easy to implement interface that can use current chipset technologies from the host processor bus, has good performance, industry standard protocols, and provisions for system boot control (See Page 1 Section 1.0 Paragraph 5 of AP-758), and because it can be easily integrated into existing systems that support the PCI bus (See Page 1 Section 1.0 Paragraph 9 of AP-758).

36. In reference to Claim 50, Shanley, AP-758, and Lay teach the limitations as applied to Claim 45 above. Shanley, AP-758, and Lay do not teach that the machine state information stored in the non-volatile memory is in a compressed data format, and

the method further comprises decompressing the machine state information to be transferred to the computer system. Lay, however, teaches that it is well known to compress an image of the machine state information to be saved to disk, and to later decompress the image when booting the system (See Column 1 Lines 51-60).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the computer system of Shanley with the fast boot by saving the system state to non-volatile memory and the compression and decompression of the boot image of Lay in order to allow the system to boot up faster (See Column 1 Lines 39-50 and Column 2 Lines 23-40 of Lay). It would have been further obvious to place the non-volatile memory on the PC card of AP-758 in order to allow the system to function as an embedded system (See Page 1 Section 1.0 Paragraph 1 of AP-758), because the PCI bus presents an easy to implement interface that can use current chipset technologies from the host processor bus, has good performance, industry standard protocols, and provisions for system boot control (See Page 1 Section 1.0 Paragraph 5 of AP-758), and because it can be easily integrated into existing systems that support the PCI bus (See Page 1 Section 1.0 Paragraph 9 of AP-758). It would have been further obvious to compress the image of the machine state because it is well known that compressed data takes up less space in memory than uncompressed data, thus allowing more efficient use of the memory space.

37. Claims 3, 10, 18, 24, 43, and 49 are rejected under 35 U.S.C. 103(a) as being unpatentable over Shanley, AP-758, and Lay as applied to Claims 2, 9, 17, 23, 39, and 46 above, and further in view of US Patent Number 6,256,692 Yoda et al. ("Yoda").

38. In reference to Claim 3, Shanley, AP-758, and Lay teach the limitations as applied to Claim 1 above. Shanley, AP-758, and Lay do not teach a PCI-CardBus bridge coupled to the PCI bus to provide communication between the PCI bus and a CardBus compatible device coupled to the PCI-CardBus bridge. AP-758 teaches placing non-volatile memory on a removable PCI card (See Page 1 Section 1.0). Yoda teaches connecting a non-volatile memory device to a CardBus (See Column 4 Lines 64-66) and connecting the CardBus through a bridge to a PCI bus (See Figure 2 Number 24, Column 1 Lines 7-12, Column 2 Lines 13-30, and Column 4 Lines 61-63).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to construct the device of Shanley, AP-758, and Lay with the CardBus device connected to a PCI bus of Yoda, resulting in the invention of Claim 3, because CardBus devices provide the same facilities as PCI devices and allow the devices to be configured as a card of a business card size rather than for fitting onto a printed substrate as with PCI devices (See Column 1 Lines 32-38 of Yoda).

39. In reference to Claim 10, Shanley, AP-758, and Lay teach the limitations as applied to Claim 9 above. Shanley further teaches that the bus comprises a PCI bus (See Figure 2-3 'PCI Bus'). Shanley, AP-758, and Lay do not teach that the computer

system further comprises a PCI-CardBus bridge coupled to the PCI bus to provide communication between the PCI bus and a CardBus compatible device coupled to the PCI-CardBus bridge. AP-758 teaches placing non-volatile memory on a removable PCI card (See Page 1 Section 1.0). Yoda teaches connecting a non-volatile memory device to a CardBus (See Column 4 Lines 64-66) and connecting the CardBus through a bridge to a PCI bus (See Figure 2 Number 24, Column 1 Lines 7-12, Column 2 Lines 13-30, and Column 4 Lines 61-63).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to construct the device of Shanley, AP-758, and Lay with the CardBus device connected to a PCI bus of Yoda, resulting in the invention of Claim 10, because CardBus devices provide the same facilities as PCI devices and allow the devices to be configured as a card of a business card size rather than for fitting onto a printed substrate as with PCI devices (See Column 1 Lines 32-38 of Yoda).

40. In reference to Claim 18, Shanley, AP-758, and Lay teach the limitations as applied to Claim 17 above. Shanley further teaches that the second bus is a PCI bus (See Figure 2-3 'PCI Bus'). Shanley, AP-758, and Lay do not teach that the bus interface is CardBus compatible. AP-758 teaches placing non-volatile memory on a removable PCI card (See Page 1 Section 1.0). Yoda teaches connecting a non-volatile memory device to a CardBus (See Column 4 Lines 64-66) and connecting the CardBus through a bridge to a PCI bus (See Figure 2 Number 24, Column 1 Lines 7-12, Column 2 Lines 13-30, and Column 4 Lines 61-63).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to construct the device of Shanley, AP-758, and Lay with the CardBus device connected to a PCI bus of Yoda, resulting in the invention of Claim 18, because CardBus devices provide the same facilities as PCI devices and allow the devices to be configured as a card of a business card size rather than for fitting onto a printed substrate as with PCI devices (See Column 1 Lines 32-38 of Yoda).

41. In reference to Claim 24, Shanley, AP-758, and Lay teach the limitations as applied to Claim 23 above. Shanley further teaches that the bus is a PCI bus (See Figure 2-3 'PCI Bus'). Shanley, AP-758, and Lay do not teach that the bus interface is CardBus compatible. AP-758 teaches placing non-volatile memory on a removable PCI card (See Page 1 Section 1.0). Yoda teaches connecting a non-volatile memory device to a CardBus (See Column 4 Lines 64-66) and connecting the CardBus through a bridge to a PCI bus (See Figure 2 Number 24, Column 1 Lines 7-12, Column 2 Lines 13-30, and Column 4 Lines 61-63).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to construct the device of Shanley, AP-758, and Lay with the CardBus device connected to a PCI bus of Yoda, resulting in the invention of Claim 24, because CardBus devices provide the same facilities as PCI devices and allow the devices to be configured as a card of a business card size rather than for fitting onto a printed substrate as with PCI devices (See Column 1 Lines 32-38 of Yoda).

42. In reference to Claim 43, Shanley, AP-758, and Lay teach the limitations as applied to Claim 38 above. Shanley, AP-758, and Lay do not teach that transferring the machine state information to the PC card comprises transferring data from the CPU and the memory to the PC card in accordance with a CardBus protocol. AP-758 teaches placing non-volatile memory on a removable PCI card (See Page 1 Section 1.0). Yoda teaches connecting a non-volatile memory device to a CardBus (See Column 4 Lines 64-66) and connecting the CardBus through a bridge to a PCI bus (See Figure 2 Number 24, Column 1 Lines 7-12, Column 2 Lines 13-30, and Column 4 Lines 61-63).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to construct the device of Shanley, AP-758, and Lay with the CardBus device connected to a PCI bus of Yoda, resulting in the invention of Claim 43, because CardBus devices provide the same facilities as PCI devices and allow the devices to be configured as a card of a business card size rather than for fitting onto a printed substrate as with PCI devices (See Column 1 Lines 32-38 of Yoda).

43. In reference to Claim 49, Shanley, AP-758, and Lay teach the limitations as applied to Claim 45 above. Shanley, AP-758, and Lay do not teach that transferring the machine state information from the non-volatile memory comprises transferring data from the PC card to the computer system in accordance with a CardBus protocol. AP-758 teaches placing non-volatile memory on a removable PCI card (See Page 1 Section 1.0). Yoda teaches connecting a non-volatile memory device to a CardBus (See Column 4 Lines 64-66) and connecting the CardBus through a bridge to a PCI bus (See

Figure 2 Number 24, Column 1 Lines 7-12, Column 2 Lines 13-30, and Column 4 Lines 61-63).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to construct the device of Shanley, AP-758, and Lay with the CardBus device connected to a PCI bus of Yoda, resulting in the invention of Claim 49, because CardBus devices provide the same facilities as PCI devices and allow the devices to be configured as a card of a business card size rather than for fitting onto a printed substrate as with PCI devices (See Column 1 Lines 32-38 of Yoda).

44. Claims 28, 29, 30, 31, 32, 33, 34, 35, 36, and 37 are rejected under 35 U.S.C. 103(a) as being unpatentable over Shanley, Lay, AP-758, and Yoda.

45. In reference to Claim 28, Shanley teaches a computer system having a central processing unit (CPU) (See Figure 2-3 'CPU') coupled to a memory (See Figure 2-3 'Main Memory') via a CPU bus (See Figure 2-3 'CPU Local Bus'), and further having a PCI bus coupled to the CPU bus to provide communication with the CPU and the memory (See Figure 2-3 'PCI Bus'); and a PC card comprising an interface coupled to the PCI bus for transferring data thereto and therefrom (See Figure 2-3). Shanley does not teach that the PC card is a CardBus compatible PC card for restoring a machine state of a computer system, the PC card having a non-volatile memory coupled to the interface for storing and providing machine state information corresponding to the machine state; a controller coupled to the interface and non-volatile memory to control

the storing of machine state date in the non-volatile memory and the retrieval of machine state information from the non-volatile memory; and a transfer component for directing the controller to coordinate with the CPU access to the non-volatile memory and the memory to store and download the machine state information for capturing and restoring, respectively, a corresponding machine state of a computer system. AP-758 teaches a PC card having a non-volatile memory (See Page 1 Section 1.0). Lay teaches storing machine state information in a non-volatile memory (See Figure 4 and Column 2 Line 23 – Column 3 Line 5) and a controller coupled to the non-volatile memory for coordinating with the CPU access to the non-volatile memory and the memory to store and download the machine state information for capturing and restoring, respectively, a corresponding machine state of a computer system (See Column 2 Lines 26-31, Column 2 Lines 45-54, and Column 3 Lines 2-5); and a transfer component directing the controller to coordinate with the CPU access to the non-volatile memory and the memory to store and download the machine state information for capturing and restoring, respectively, a corresponding machine state of a computer system (See Figures 4 and 6, Column 2 Lines 26-31, Column 2 Lines 45-54, Column 3 Lines 2-5, and Column 5 Lines 39-63). Yoda teaches connecting a non-volatile memory device to a CardBus (See Column 4 Lines 64-66) and connecting the CardBus through a bridge to a PCI bus (See Figure 2 Number 24, Column 1 Lines 7-12, Column 2 Lines 13-30, and Column 4 Lines 61-63).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the computer system of Shanley with the fast boot by

saving the system state to non-volatile memory of Lay in order to allow the system to boot up faster (See Column 1 Lines 39-50 and Column 2 Lines 23-40 of Lay). It would have been further obvious to place the non-volatile memory on the PC card of AP-758 in order to allow the system to function as an embedded system (See Page 1 Section 1.0 Paragraph 1 of AP-758), because the PCI bus presents an easy to implement interface that can use current chipset technologies from the host processor bus, has good performance, industry standard protocols, and provisions for system boot control (See Page 1 Section 1.0 Paragraph 5 of AP-758), and because it can be easily integrated into existing systems that support the PCI bus (See Page 1 Section 1.0 Paragraph 9 of AP-758). It would have been further obvious to make the PC card a CardBus device connected to a PCI bus of Yoda because CardBus devices provide the same facilities as PCI devices and allows the devices to be configured as a card of a business card size rather than for fitting onto a printed substrate as with PCI devices (See Column 1 Lines 32-38 of Yoda).

46. In reference to Claim 29, Shanley, AP-758, Lay, and Yoda teach the limitations as applied to Claim 28 above. Lay further teaches that the machine state information comprises data from the memory and CPU for returning the computer system to the same condition of operability as when the machine state information was stored in the non-volatile memory (See Figure 3 and Column 2 Lines 26-37).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the computer system of Shanley with the fast boot by

saving the system state to non-volatile memory of Lay in order to allow the system to boot up faster (See Column 1 Lines 39-50 and Column 2 Lines 23-40 of Lay). It would have been further obvious to place the non-volatile memory on the PC card of AP-758 in order to allow the system to function as an embedded system (See Page 1 Section 1.0 Paragraph 1 of AP-758), because the PCI bus presents an easy to implement interface that can use current chipset technologies from the host processor bus, has good performance, industry standard protocols, and provisions for system boot control (See Page 1 Section 1.0 Paragraph 5 of AP-758), and because it can be easily integrated into existing systems that support the PCI bus (See Page 1 Section 1.0 Paragraph 9 of AP-758). It would have been further obvious to make the PC card a CardBus device connected to a PCI bus of Yoda because CardBus devices provide the same facilities as PCI devices and allows the devices to be configured as a card of a business card size rather than for fitting onto a printed substrate as with PCI devices (See Column 1 Lines 32-38 of Yoda).

47. In reference to Claim 30, Shanley, AP-758, Lay, and Yoda teach the limitations as applied to Claim 1 above. AP-758 further teaches that the non-volatile memory of the PC card comprises a flash memory device (See Page 1 Section 1.0 Paragraphs 2-7).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the computer system of Shanley with the fast boot by saving the system state to non-volatile memory of Lay in order to allow the system to

boot up faster (See Column 1 Lines 39-50 and Column 2 Lines 23-40 of Lay). It would have been further obvious to place the non-volatile memory on the PC card of AP-758 in order to allow the system to function as an embedded system (See Page 1 Section 1.0 Paragraph 1 of AP-758), because the PCI bus presents an easy to implement interface that can use current chipset technologies from the host processor bus, has good performance, industry standard protocols, and provisions for system boot control (See Page 1 Section 1.0 Paragraph 5 of AP-758), and because it can be easily integrated into existing systems that support the PCI bus (See Page 1 Section 1.0 Paragraph 9 of AP-758). It would have been further obvious to make the PC card a CardBus device connected to a PCI bus of Yoda because CardBus devices provide the same facilities as PCI devices and allows the devices to be configured as a card of a business card size rather than for fitting onto a printed substrate as with PCI devices (See Column 1 Lines 32-38 of Yoda).

48. In reference to Claim 31, Shanley, AP-758, Lay, and Yoda teach the limitations as applied to Claim 28 above. Lay further teaches the transfer component comprises: a storing component for directing the controller to store machine state information from the CPU and memory to the non-volatile memory (See Figure 3, Column 2 Lines 26-31, and Column 2 Lines 45-54); and a download component for directing the controller to transfer data from the nonvolatile memory to the CPU and the memory (See Figure 5, Column 3 Lines 2-5, and Column 5 Lines 32-63).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the computer system of Shanley with the fast boot by saving the system state to non-volatile memory of Lay in order to allow the system to boot up faster (See Column 1 Lines 39-50 and Column 2 Lines 23-40 of Lay). It would have been further obvious to place the non-volatile memory on the PC card of AP-758 in order to allow the system to function as an embedded system (See Page 1 Section 1.0 Paragraph 1 of AP-758), because the PCI bus presents an easy to implement interface that can use current chipset technologies from the host processor bus, has good performance, industry standard protocols, and provisions for system boot control (See Page 1 Section 1.0 Paragraph 5 of AP-758), and because it can be easily integrated into existing systems that support the PCI bus (See Page 1 Section 1.0 Paragraph 9 of AP-758). It would have been further obvious to make the PC card a CardBus device connected to a PCI bus of Yoda because CardBus devices provide the same facilities as PCI devices and allows the devices to be configured as a card of a business card size rather than for fitting onto a printed substrate as with PCI devices (See Column 1 Lines 32-38 of Yoda).

49. In reference to Claim 32, Shanley, AP-758, Lay, and Yoda teach the limitations as applied to Claim 28 above. Shanley, AP-758, Lay, and Yoda do not teach compression and decompression components for compressing the machine state information to be stored and decompressing the stored compressed machine state information to be downloaded, respectively. Lay, however, teaches that it is well known

to compress an image of the machine state information to be saved to disk, and to later decompress the image when booting the system (See Column 1 Lines 51-60).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the computer system of Shanley with the fast boot by saving the system state to non-volatile memory of Lay in order to allow the system to boot up faster (See Column 1 Lines 39-50 and Column 2 Lines 23-40 of Lay). It would have been further obvious to place the non-volatile memory on the PC card of AP-758 in order to allow the system to function as an embedded system (See Page 1 Section 1.0 Paragraph 1 of AP-758), because the PCI bus presents an easy to implement interface that can use current chipset technologies from the host processor bus, has good performance, industry standard protocols, and provisions for system boot control (See Page 1 Section 1.0 Paragraph 5 of AP-758), and because it can be easily integrated into existing systems that support the PCI bus (See Page 1 Section 1.0 Paragraph 9 of AP-758). It would have been further obvious to make the PC card a CardBus device connected to a PCI bus of Yoda because CardBus devices provide the same facilities as PCI devices and allows the devices to be configured as a card of a business card size rather than for fitting onto a printed substrate as with PCI devices (See Column 1 Lines 32-38 of Yoda). It would have been further obvious to compress the image of the machine state because it is well known that compressed data takes up less space in memory than uncompressed data, thus allowing more efficient use of the memory space.

50. In reference to Claim 33, Shanley teaches a computer system comprising a central processing unit (CPU) (See Figure 2-3 'CPU'); a local CPU bus coupled to the CPU (See Figure 2-3 'CPU Local Bus'); a memory coupled to the local CPU bus to store data accessible by the CPU via the local CPU bus (See Figure 2-3 'Main Memory'); a PCI bus coupled to the local CPU bus to provide communication with the CPU and the memory via the local CPU bus (See Figure 2-3 'PCI Bus'); and a PC card (See Figure 2-3). Shanley does not teach a PCI-CardBus bridge coupled to the PCI bus to provide communication between the PCI bus and a CardBus compatible device; a CardBus compatible PC card coupled to PCI-CardBus bridge, the PC card having a non-volatile memory for storing machine state information corresponding to the machine state, and further having a controller coupled to the non-volatile memory to control the storing of data therein and the retrieval of data therefrom; and a transfer component for directing the controller to coordinate with the CPU access to the non-volatile memory and the memory to store and download the machine state information for capturing and restoring, respectively, a corresponding machine state of a computer system. AP-758 teaches a PC card having a non-volatile memory (See Page 1 Section 1.0). Lay teaches storing machine state information in a non-volatile memory (See Figure 4 and Column 2 Line 23 – Column 3 Line 5); controller coupled to the non-volatile memory to control the storing of data therein and the retrieval of data therefrom (See Column 2 Lines 26-31, Column 2 Lines 45-54, and Column 3 Lines 2-5); and a transfer component for directing the controller to coordinate with the CPU access to the non-volatile memory and the memory to store and download the machine state information

for capturing and restoring, respectively, a corresponding machine state of a computer system (See Figures 4 and 6, Column 2 Lines 26-31, Column 2 Lines 45-54, Column 3 Lines 2-5, and Column 5 Lines 39-63). Yoda teaches connecting a non-volatile memory device to a CardBus (See Column 4 Lines 64-66) and connecting the CardBus through a bridge to a PCI bus (See Figure 2 Number 24, Column 1 Lines 7-12, Column 2 Lines 13-30, and Column 4 Lines 61-63).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the computer system of Shanley with the fast boot by saving the system state to non-volatile memory of Lay in order to allow the system to boot up faster (See Column 1 Lines 39-50 and Column 2 Lines 23-40 of Lay). It would have been further obvious to place the non-volatile memory on the PC card of AP-758 in order to allow the system to function as an embedded system (See Page 1 Section 1.0 Paragraph 1 of AP-758), because the PCI bus presents an easy to implement interface that can use current chipset technologies from the host processor bus, has good performance, industry standard protocols, and provisions for system boot control (See Page 1 Section 1.0 Paragraph 5 of AP-758), and because it can be easily integrated into existing systems that support the PCI bus (See Page 1 Section 1.0 Paragraph 9 of AP-758). It would have been further obvious to make the PC card a CardBus device connected to a PCI bus of Yoda because CardBus devices provide the same facilities as PCI devices and allows the devices to be configured as a card of a business card size rather than for fitting onto a printed substrate as with PCI devices (See Column 1 Lines 32-38 of Yoda).

51. In reference to Claim 34, Shanley, AP-758, Lay, and Yoda teach the limitations as applied to Claim 33 above. AP-758 further teaches that the non-volatile memory of the PC card comprises a flash memory device (See Page 1 Section 1.0 Paragraphs 2-7).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the computer system of Shanley with the fast boot by saving the system state to non-volatile memory of Lay in order to allow the system to boot up faster (See Column 1 Lines 39-50 and Column 2 Lines 23-40 of Lay). It would have been further obvious to place the non-volatile memory on the PC card of AP-758 in order to allow the system to function as an embedded system (See Page 1 Section 1.0 Paragraph 1 of AP-758), because the PCI bus presents an easy to implement interface that can use current chipset technologies from the host processor bus, has good performance, industry standard protocols, and provisions for system boot control (See Page 1 Section 1.0 Paragraph 5 of AP-758), and because it can be easily integrated into existing systems that support the PCI bus (See Page 1 Section 1.0 Paragraph 9 of AP-758). It would have been further obvious to make the PC card a CardBus device connected to a PCI bus of Yoda because CardBus devices provide the same facilities as PCI devices and allows the devices to be configured as a card of a business card size rather than for fitting onto a printed substrate as with PCI devices (See Column 1 Lines 32-38 of Yoda).

52. In reference to Claim 35, Shanley, AP-758, Lay, and Yoda teach the limitations as applied to Claim 33 above. AP-758 further teaches that the PC card further includes a bus interface coupled to the PCI bus, and further coupled to the non-volatile memory and the controller to transfer data between the memory and the PCI bus in accordance with the PCI data format and transfer protocol (See Pages 1-2 and Figures 1-2).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the computer system of Shanley with the fast boot by saving the system state to non-volatile memory of Lay in order to allow the system to boot up faster (See Column 1 Lines 39-50 and Column 2 Lines 23-40 of Lay). It would have been further obvious to place the non-volatile memory on the PC card of AP-758 in order to allow the system to function as an embedded system (See Page 1 Section 1.0 Paragraph 1 of AP-758), because the PCI bus presents an easy to implement interface that can use current chipset technologies from the host processor bus, has good performance, industry standard protocols, and provisions for system boot control (See Page 1 Section 1.0 Paragraph 5 of AP-758), and because it can be easily integrated into existing systems that support the PCI bus (See Page 1 Section 1.0 Paragraph 9 of AP-758). It would have been further obvious to make the PC card a CardBus device connected to a PCI bus of Yoda because CardBus devices provide the same facilities as PCI devices and allows the devices to be configured as a card of a business card size rather than for fitting onto a printed substrate as with PCI devices (See Column 1 Lines 32-38 of Yoda).

53. In reference to Claim 36, Shanley, AP-758, Lay, and Yoda teach the limitations as applied to Claim 33 above. Shanley, AP-758, Lay, and Yoda do not teach compression and decompression components for compressing the machine state information to be stored and decompressing the stored compressed machine state information to be downloaded, respectively. Lay, however, teaches that it is well known to compress an image of the machine state information to be saved to disk, and to later decompress the image when booting the system (See Column 1 Lines 51-60).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the computer system of Shanley with the fast boot by saving the system state to non-volatile memory of Lay in order to allow the system to boot up faster (See Column 1 Lines 39-50 and Column 2 Lines 23-40 of Lay). It would have been further obvious to place the non-volatile memory on the PC card of AP-758 in order to allow the system to function as an embedded system (See Page 1 Section 1.0 Paragraph 1 of AP-758), because the PCI bus presents an easy to implement interface that can use current chipset technologies from the host processor bus, has good performance, industry standard protocols, and provisions for system boot control (See Page 1 Section 1.0 Paragraph 5 of AP-758), and because it can be easily integrated into existing systems that support the PCI bus (See Page 1 Section 1.0 Paragraph 9 of AP-758). It would have been further obvious to make the PC card a CardBus device connected to a PCI bus of Yoda because CardBus devices provide the same facilities as PCI devices and allows the devices to be configured as a card of a business card size rather than for fitting onto a printed substrate as with PCI devices

(See Column 1 Lines 32-38 of Yoda). It would have been further obvious to compress the image of the machine state because it is well known that compressed data takes up less space in memory than uncompressed data, thus allowing more efficient use of the memory space.

54. In reference to Claim 37, Shanley, AP-758, Lay, and Yoda teach the limitations as applied to Claim 33 above. Lay further teaches that the machine state information comprises data from the memory and CPU for returning the computer system to the same condition of operability as when the machine state information was stored in the non-volatile memory (See Figure 3 and Column 2 Lines 26-37).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the computer system of Shanley with the fast boot by saving the system state to non-volatile memory of Lay in order to allow the system to boot up faster (See Column 1 Lines 39-50 and Column 2 Lines 23-40 of Lay). It would have been further obvious to place the non-volatile memory on the PC card of AP-758 in order to allow the system to function as an embedded system (See Page 1 Section 1.0 Paragraph 1 of AP-758), because the PCI bus presents an easy to implement interface that can use current chipset technologies from the host processor bus, has good performance, industry standard protocols, and provisions for system boot control (See Page 1 Section 1.0 Paragraph 5 of AP-758), and because it can be easily integrated into existing systems that support the PCI bus (See Page 1 Section 1.0 Paragraph 9 of AP-758). It would have been further obvious to make the PC card a

CardBus device connected to a PCI bus of Yoda because CardBus devices provide the same facilities as PCI devices and allows the devices to be configured as a card of a business card size rather than for fitting onto a printed substrate as with PCI devices (See Column 1 Lines 32-38 of Yoda).

Response to Arguments

55. Applicant's arguments filed 3 March 2005 have been fully considered but they are not persuasive.

56. In response to Applicant's argument that the combination of Shanley, AP-758, and Lay do not teach "a controller coupled to the non-volatile memory for coordinating with the CPU access to the non-volatile memory and the memory to store and download the machine state information for capturing and restoring, respectively, a corresponding machine state of a computer system", the Examiner notes that AP-758 teaches a PC card containing a non-volatile memory and a controller (See Page 2 Figure 2); and Lay teaches a controller coupled to a non-volatile memory for coordinating with the CPU access to the non-volatile memory and the memory to store and download the machine state information for capturing and restoring, respectively, a corresponding machine state of a computer system (See Column 2 Lines 26-31, Column 2 Lines 45-54, Column 3 Lines 2-5, and Column 6 Lines 36-43).

57. In response to Applicant's argument that all the transfer features for storing boot images are taught by Lay are executed from an application, not from a controller, the Examiner notes that Lay teaches that "although the various methods described are conveniently implemented in a general purpose computer selectively activated or reconfigured by software, one or ordinary skill in the art would also recognize that such methods may be carried out in hardware, in firmware, or in more specialized apparatus constructed to perform the required method steps" (See Column 6 Lines 37-43). The Examiner further notes that it is well known in the art that hardware and software are logically equivalent, as evidenced by Structured Computer Organization by Andrew S. Tanenbaum ("Tanenbaum") (See Page 1).

58. In response to Applicant's argument that Lay does not teach that the machine state information should be stored in a non-volatile memory on a PC card, the Examiner notes that, as shown in the above rejections, the combination of Lay and AP-758 was relied upon to teach this limitation. Lay teaches storing a boot image in non-volatile memory (See Column 1 Lines 61-62 and Column 2 Lines 23-45). AP-758 teaches the benefits of locating a non-volatile memory (and thus, the boot image) on a PC card (See Page 1).

59. In response to Applicant's argument that the Examiner's conclusion of obviousness is based upon improper hindsight reasoning, it must be recognized that any judgment on obviousness is in a sense necessarily a reconstruction based upon

hindsight reasoning. But so long as it takes into account only knowledge which was within the level of ordinary skill at the time the claimed invention was made, and does not include knowledge gleaned only from the Applicant's disclosure, such a reconstruction is proper. See *In re McLaughlin*, 443 F.2d 1392, 170 USPQ 209 (CCPA 1971).

60. In response to Applicant's argument that there is no suggestion to combine the references, the Examiner recognizes that obviousness can only be established by combining or modifying the teachings of the prior art to produce the claimed invention where there is some teaching, suggestion, or motivation to do so found either in the references themselves or in the knowledge generally available to one of ordinary skill in the art. See *In re Fine*, 837 F.2d 1071, 5 USPQ2d 1596 (Fed. Cir. 1988) and *In re Jones*, 958 F.2d 347, 21 USPQ2d 1941 (Fed. Cir. 1992). In this case, it would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the computer system of Shanley with the fast boot by saving the system state to non-volatile memory of Lay in order to allow the system to boot up faster (See Column 1 Lines 39-50 and Column 2 Lines 23-40 of Lay). It would have been further obvious to place the non-volatile memory on the PC card of AP-758 in order to allow the system to function as an embedded system (See Page 1 Section 1.0 Paragraph 1 of AP-758), because the PCI bus presents an easy to implement interface that can use current chipset technologies from the host processor bus, has good performance, industry standard protocols, and provisions for system boot control (See Page 1 Section

1.0 Paragraph 5 of AP-758), and because it can be easily integrated into existing systems that support the PCI bus (See Page 1 Section 1.0 Paragraph 9 of AP-758).

61. In response to Applicant's argument that Lay does not teach storing machine state information anywhere but in disk storage, and that there is nothing in Lay that teaches or suggests that the memory that holds the machine state information should be a portable memory, the Examiner notes that Lay teaches storing the boot image in "a hard disk of other non-volatile storage" (See Column 2 Lines 44-45). When constructing the device of Lay, one of ordinary skill in the art would naturally look to various types on non-volatile storage, including the PC card flash memory of AP-758. One would be motivated to place the boot image of Lay on the PC card flash memory of AP-758 because it interfaces to a PCI bus, and thus can be easily integrated into existing systems that support the PCI bus (See Page 1 Section 1.0 Paragraph 9) and because it presents an easy to implement interface that can use current chipset technologies from the host processor bus, has good performance, industry standard protocols, and provisions for system boot control (See Page 1 Section 1.0 Paragraph 5 of AP-758). One would further be motivated to place the boot image of Lay on the PC card flash memory of AP-758 because it would allow the device of Lay to be used in embedded systems (See Page 1 Section 1.0 Paragraph 1 of AP-758). Further, because the device of Lay is useful for mission critical computer performance (See Column 1 Lines 39-50 of Lay), one of ordinary skill in the art would naturally look to such a device when constructing an embedded system, which is expected to function for long periods of

time without human intervention (See 'embedded system' in The Free On-Line Dictionary of Computing ("FOLDOC")).

62. In response to Applicant's argument that one of ordinary skill in the art might be motivated away from using PC cards and flash memory to store machine state data, the Examiner notes that because the PC card of AP-758 is designed to be used in an embedded system, it would not be removed unless the system is being repaired or upgraded. Thus, the PC card flash memory is being used as a permanent non-volatile memory as opposed to a portable non-volatile memory, which makes it an ideal location for a mission critical boot image of the system.

Information Disclosure Statement

63. The information disclosure statements filed 22 November 2004, 28 March 2005, and 11 April 2005 fail to comply with 37 CFR 1.98(a)(1), which requires the following: (1) a list of all patents, publications, applications, or other information submitted for consideration by the Office; (2) U.S. patents and U.S. patent application publications listed in a section separately from citations of other documents; (3) the application number of the application in which the information disclosure statement is being submitted on each page of the list; (4) a column that provides a blank space next to each document to be considered, for the examiner's initials; and (5) a heading that clearly indicates that the list is an information disclosure statement. The information

disclosure statement has been placed in the application file, but the information referred to therein has not been considered. A list of the patent applications having a column that provides a blank space next to each document to be considered for the examiner's initials was not provided.

64. Applicant's assistance is requested in identifying any prior art believed to be of particular relevance to the current application.

Conclusion

65. The following prior art made of record and not relied upon is considered pertinent to Applicant's disclosure: US Patent Number 6,463,509 to Teoman et al.

66. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of

the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the Examiner should be directed to Thomas J. Cleary whose telephone number is 571-272-3624. The Examiner can normally be reached on Monday-Thursday (7-3:30), Alt. Fridays (7-2:30).

If attempts to reach the Examiner by telephone are unsuccessful, the Examiner's supervisor, Mark H. Rinehart can be reached on 571-272-3632. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

TJC

Khanh Dang
Primary Examiner

Thomas J. Cleary
Patent Examiner
Art Unit 2111